



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/849,736	05/04/2001	G. Glenn Henry	CNTR:2021	8084
23669	7590	01/25/2005	EXAMINER	
HUFFMAN LAW GROUP, P.C. 1832 N. CASCADE AVE. COLORADO SPRINGS, CO 80907-7449				HUISMAN, DAVID J
ART UNIT		PAPER NUMBER		
				2183

DATE MAILED: 01/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/849,736	HENRY ET AL.	
	Examiner David J. Huisman	Art Unit 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the corresponding address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 29 November 2004.
 2a) This action is FINAL. 2b) This action is non-final.
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 31-38,43-68 and 70 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
 5) Claim(s) _____ is/are allowed.
 6) Claim(s) 31-38,43-68 and 70 is/are rejected.
 7) Claim(s) _____ is/are objected to.
 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
 10) The drawing(s) filed on 04 May 2001 is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date: _____ |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>11/10/04, 12/15/04</u> . | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 31-38, 43-68, and 70 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: IDS as received on 11/10/2004, Amendment as received on 11/29/2004, and IDS as received on 12/15/2004. The Non-Patent Literature cited in the IDS filed on 12/15/2004, is not currently part of the electronic file. Consequently, the NPL for this IDS has not been considered.

Claim Objections

3. Claim 48 is objected to because of the following informalities: In line 2, insert either --the-- or --said-- before "instruction buffer". Appropriate correction is required.
4. Claim 49 is objected to because of the following informalities: In line 2, insert either --the-- or --said-- before "instruction buffer". Appropriate correction is required.

Withdrawn Rejections

5. Applicant has overcome, via amendment, the prior art rejections set forth in the previous Office Action for claims 31 (and its dependent claims) and claim 66 (and its dependent claims). Consequently, these rejections are hereby withdrawn by the examiner. However, upon further consideration, a new grounds of rejection is applied below.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claim 31 recites the limitation "said indicator" in line 10. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator". Therefore, if there are multiple indicators (plural), there is no basis for "said indicator" (singular).

8. Claim 31 recites the limitation "said instruction" in line 11. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "said instructions". The examiner asserts that "each of said instructions" does not provide antecedent basis for "said instruction".

9. Claim 32 recites the limitation "one of said plurality of indicators" in line 2. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator". Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

10. Claim 43 recites the limitation "said plurality of indicators" in lines 5, 6, and 12. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

11. Claim 44 recites the limitation "said plurality of indicators" in lines 5 and 6. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at

Art Unit: 2183

least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

12. Claim 45 recites the limitation "said plurality of indicators" in lines 4 and 5. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

13. Claim 46 recites the limitation "said plurality of indicators" in lines 2 and 4. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

14. Claim 47 recites the limitation "said plurality of indicators" in lines 2 and 4. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

15. Claim 48 recites the limitation "said plurality of indicators" in lines 6 and 7. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

16. Claim 49 recites the limitation "said plurality of indicators" in lines 6 and 7. There is insufficient antecedent basis for this limitation in the claim, as applicant previously claims "at least one indicator" in claim 31. Therefore, if there is a single indicator (singular), there is no basis for "said plurality of indicators" (plural).

Claim Rejections - 35 USC § 102

17. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

18. Claims 31-33 are rejected under 35 U.S.C. 102(b) as being anticipated by Black et al., U.S. Patent No. 5,761,723 (as disclosed by applicant and herein referred to as Black).

19. Referring to claim 31, Black has taught a pipelined microprocessor comprising:

a) an instruction cache (Fig.1, component 14) that is indexed by a fetch address, said instruction cache for caching instructions, and for providing said instructions to an instruction buffer for storage therein (see Fig.1, component 18, which is shown in more detail in Fig.3, and note that the sequencer comprises an instruction buffer, and it therefore is an instruction buffer).

b) a branch target address cache (Fig.3, component 48), coupled to said instruction buffer and indexed by said fetch address, for caching branch target addresses of previously executed branch instructions. See column 1, lines 44-50.

c) said instruction buffer comprising at least one indicator associated with each of said instructions, wherein said indicator has a true value if said branch target address cache predicts said instruction is one of said previously executed branch instructions and the microprocessor has speculatively branched to one of said branch target addresses cached for said one of said previously executed branch instructions. See column 8, lines 12-18, and note that for each instruction in the instruction buffer, a HIT?MISS indicator is generated from the BTAC. If true,

Art Unit: 2183

i.e., a HIT occurred in the BTAC, then the associated instruction was predicted to be a branch and speculative branching has occurred to its target address.

20. Referring to claim 32, Black has taught a microprocessor as described in claim 31. Black has further taught that said instruction buffer includes one of said plurality of indicators associated with each byte of said each of said instructions stored in said instruction buffer. See column 8, lines 12-18, and note that the BTAC is checked for every instruction to see if it is a branch. Therefore, each instruction would have an associated HIT?MISS indicator.

21. Referring to claim 33, Black has taught a microprocessor as described in claim 31. Black has further taught that said instruction cache and said branch target address cache are accessed substantially in parallel. See column 1, lines 48-50.

Claim Rejections - 35 USC § 103

22. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

23. Claims 34-36, 50-55, and 59-65 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell et al., U.S. Patent No. 5,850,543 (as applied in the previous Office Action and herein referred to as Shiell) in view of Hsu et al., U.S. Patent No. 5,948,100 (as applied in the previous Office Action and herein referred to as Hsu).

24. Referring to claim 34, Shiell has taught a method of speculatively branching in a pipelined microprocessor, comprising:

Art Unit: 2183

- a) caching, in a branch target address cache (BTAC), a plurality of branch target addresses of previously executed branch instructions. See Fig. 2, BTB 56 and column 7, lines 40-42.
- b) Shiell has not taught caching a direct indication of whether each of the branch instructions spans more than one instruction cache line. However, Hsu has taught a BTB which caches the length of each branch instruction. The length is used to determine whether or not a branch spans a cache line by adding it to the branch instruction address. See column 13, lines 18-30. The examiner asserts that the length is a direct indication **with respect to the instruction address**. That is, given the instruction address, all that is needed is the length to make a determination as to whether an associated instructions spans cache lines. For instance, if a cache line holds four bytes, and a branch instruction begins in the first byte, a length of two directly indicates that the instruction does not span multiple cache lines. On the other hand, if the branch instruction begins in the last cache line byte, a length of two directly indicates that it does span multiple cache lines. By tracking the length, the system is able to accommodate branch instructions that cross multiple cache blocks, which is useful in variable length instruction systems, as in both Hsu and Shiell (see the title of Hsu and column 6, lines 32-35 of Shiell). Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to cache an indication of whether a branch instruction spans an instruction cache line.
- c) accessing said BTAC with a fetch address of an instruction cache after said caching (fig. 2 indicates that the fetch address FA is used to access the BTB 56; col. 8, lines 13-16 indicate that the accessing is done when branch history is stored in the BTB).
- d) determining whether said fetch address hits in said BTAC in response to said accessing (col. 8, lines 13-16).

e) branching the microprocessor to one of said plurality of branch target addresses selected by said fetch address if said fetch address hits in said BTAC whether or not a branch instruction is cached in a line of said instruction cache indexed by said fetch address (col. 7, lines 40-45 indicate that the target address is used to generate the fetch address during the fetch stage before the decoding the instruction without knowledge of whether the instruction at that fetch address in the instruction cache is still a branch or not i.e. speculative execution [col. 8, lines 13-16] because the BTB and the instruction cache are accessed in parallel using the fetch address FA).

25. Referring to claim 35, Shiell in view of Hsu has taught a method as described in claim 34. Shiell has further taught storing a branch direction prediction associated with each of said plurality of branch target addresses prior to said accessing said BTAC (fig. 3, "HIS_n" field; col. 8, lines 57-67).

26. Referring to claim 36, Shiell in view of Hsu has taught a method as described in claim 35. Shiell has further taught that said branching the microprocessor to said one of said plurality of branch target addresses selected by said fetch address is performed only if said associated branch direction prediction indicates said branch instruction will be taken (Although not explicitly mentioned, the limitation is deemed inherent to the correct functioning of the method because the purpose of the prediction information when indicating that the speculative branch is taken is for instructing the processor to branch to the target address of the branch and not the next sequential address).

27. Referring to claim 50, Shiell has taught a branch target address cache (BTAC) (fig. 2, Branch Target Buffer, BTB 56) for providing a speculative target address (col. 7, lines 40-42; the target address is speculative because it is provided during the instruction fetch stage before it is

known whether the instruction is a branch or not) to address selection logic (fig. 2, multiplexers 57, 58 and 52), the address selection logic selecting a fetch address for addressing a line in an instruction cache (fig. 2 shows that the address selection logic addresses the instruction cache 16; and col. 6, lines 16-23 indicate that the fetch address addresses a stream of instruction data i.e. a line in the instruction cache), the BTAC providing the speculative target address based on a presumption that a branch instruction is present in the cache (fig. 2 shows that the BTB outputs a target address D0-D127 indexed by the fetch address without knowing whether or not a branch is present in the line of the instruction cache addressed by the fetch address because the BTB and the instruction cache are accessed in parallel using the fetch address FA), the BTAC comprising:

- a) an array of storage elements, configured to cache target addresses of previously executed branch instructions (fig. 2, BTB 56; col. 7, lines 40-42).
- b) Shiell has not taught storing speculative branch information comprising a direct indication of whether the branch instruction presumed present in the cache line spans more than one line in the instruction cache. However, Hsu has taught a BTB which caches the length of each branch instruction. The length is used to determine whether or not a branch spans a cache line by adding it to the branch instruction address. See column 13, lines 18-30. The examiner asserts that the length is a direct indication **with respect to the instruction address**. That is, given the instruction address, all that is needed is the length to make a determination as to whether an associated instruction spans cache lines. For instance, if a cache line holds four bytes, and a branch instruction begins in the first byte, a length of two directly indicates that the instruction does not span multiple cache lines. On the other hand, if the branch instruction begins in the last cache line byte, a length of two directly indicates that it does span multiple cache lines. By

tracking the length, the system is able to accommodate branch instructions that cross multiple cache blocks, which is useful in variable length instruction systems, as in both Hsu and Shiell (see the title of Hsu and column 6, lines 32-35 of Shiell). Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to cache an indication of whether a branch instruction spans an instruction cache line.

- c) an input, coupled to said array, for receiving the fetch address, to index into said array of storage elements to select one of said target addresses (fig. 2; col. 8, lines 12-16).
- d) an output, coupled to said array, for providing said one of said target addresses indexed by the fetch address to the address selection logic (fig. 2 shows the target address D0-D127 are connected to an output to the address selection logic [multiplexers 57, 58 and 52]);
- e) wherein said output provides said one of said target addresses to the address selection logic for selection as a subsequent fetch address whether or not a branch instruction is present in the line of the instruction cache addressed by the fetch address (fig. 2 shows that the BTB outputs a target address D0-D127 indexed by the fetch address to the address selection logic (multiplexers 57, 58 and 52) without knowing whether or not a branch is present in the line of the instruction cache addressed by the fetch address because the BTB and the instruction cache are accessed in parallel using the fetch address FA).

28. Referring to claim 51, Shiell in view of Hsu has taught a BTAC as described in claim 1. Shiell has further taught a second output, coupled to said array, for providing a portion of said speculative branch information to control logic for controlling the address selection logic in response to said portion of said speculative branch information (although not shown, it is deemed inherent to the BTB to have a second output which outputs a portion of the history information

indicating the outcome of the branch to control the address selection logic multiplexers. This is because when the branch is indicated in the BTB as not taken, the BTB target address should not be selected and the next sequential address should be selected).

29. Referring to claim 52, Shiell in view of Hsu has taught a BTAC as described in claim 1. Shiell has further taught that said speculative branch information comprises information predicting whether the branch instruction presumed present in the cache line will be taken (fig. 3, “HIS” field indicates that the branch will be taken if set to ‘111’ or ‘110’ [col. 8, lines 57-67]).

30. Referring to claim 53, Shiell in view of Hsu has taught a BTAC as described in claim 4. Shiell has further taught that said information predicting whether the presumed branch instruction will be taken comprises a taken/not taken bit (fig. 9, when a conditional branch, the 2nd bit in the “HIS” field is a taken/not taken bit because when set to “1” it indicates taken, when set to “0” it indicates not taken).

31. Referring to claim 54, Shiell in view of Hsu has taught a BTAC as described in claim 4. Shiell has further taught that said information predicting whether the presumed branch instruction will be taken comprises a plurality of bits (fig. 3, “HIS” field).

32. Referring to claim 55, Shiell in view of Hsu has taught a BTAC as described in claim 6. Shiell has further taught that said plurality of bits is stored in a saturating up/down counter (col. 9, lines 5-18).

33. Referring to claim 59, Shiell in view of Hsu has taught a BTAC as described in claim 1. Shiell has further taught that said speculative branch information comprises information specifying a location within the cache line of the branch instruction presumed present in the cache line (fig. 3, shows the “T_n” field as an entry in the BTB. col. 8, lines 40-44 indicate that the

Art Unit: 2183

T_n field holds information specifying the location of a specific instruction within the cache line associated with the logical address LA used to index into the BTB).

34. Referring to claim 60, Shiell in view of Hsu has taught a BTAC as described in claim 50. Shiell has not taught that said speculative branch information comprises a length of the branch instruction presumed present in the cache line. However, Hsu has taught a BTB which caches the length of each branch instruction. See column 13, lines 18-30. This allows the system to accommodate branch instructions that cross multiple cache blocks, which is useful in variable length instruction systems, as both Hsu and Shiell are (see the title of Hsu and column 6, lines 32-35 of Shiell). Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to cache the length of a branch instruction so that the system may accommodate branches that span multiple cache lines.

35. Referring to claim 61, Shiell in view of Hsu has taught a BTAC as described in claim 1. Shiell has further taught that said speculative branch information comprises an indication of a type of the branch instruction presumed present in the cache line (fig. 2, "HIS" field, col. 8, lines 57-67).

36. Referring to claim 62, Shiell in view of Hsu has taught a BTAC as described in claim 13. Shiell has further taught that said indication of said type of the branch instruction indicates whether the branch instruction is a call instruction ("HIS" = 011; col. 8, lines 57-67).

37. Referring to claim 63, Shiell in view of Hsu has taught a BTAC as described in claim 13. Shiell has further taught that said indication of said type of the branch instruction indicates whether the branch instruction is a return instruction ("HIS" = 010; col. 8, lines 57-67).

Art Unit: 2183

38. Referring to claim 64, Shiell in view of Hsu has taught a BTAC as described in claim 1. Hsu has further taught that said speculative branch information comprises an indication of the branch instruction presumed present in the cache line spans more than one line in the instruction cache. See column 13, lines 18-30, and recall that Hsu has taught a BTB which caches the length of each branch instruction. The length is used to determine whether or not a branch spans a cache line by adding it to an address. Therefore, the length is at the very least an indirect or partial indication.

39. Referring to claim 65, Shiell in view of Hsu has taught a BTAC as described in claim 1. Shiell has further taught that each of said storage elements is configured to cache a plurality of target addresses (col. 8, lines 22-27 indicates that there are 4 target addresses stored per storage element).

40. Claims 37-38, 66-68, and 70 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell in view of Hsu, as applied above, and further in view of Black, as applied above.

41. Referring to claim 37, Shiell in view of Hsu has taught a method as described in claim 34. Shiell has not taught storing a discrete indication that said branching was performed if said branching is performed, and determining from said discrete indication, subsequent to said storing, that said branching was performed. However, Black has taught such a concept. See column 6, lines 34-37, and note that the HIT?MISS signal is store din a decode buffer for later use. This HIT/MISS signal, when set to HIT, indicates that a branch has been detected and branching has been performed. This signal is then used to detect if a phantom branch was encountered, and if one has been encountered, then corrective measures are taken. As a result, in

order to detect branching for a non-branch instruction, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to include Black's concept of storing an indication.

42. Referring to claim 38, Shiell in view of Hsu in view of Black has taught a method as described in claim 37. Black has further taught that said storing said indication said branching was performed comprises storing said indication in an instruction buffer. See column 6, lines 34-37, and note that it is stored in a buffer, which also holds instructions (column 7, lines 33-35).

43. Referring to claim 66, Shiell has taught a method of speculatively branching in a pipelined microprocessor (fig. 1), comprising:

- a) caching a plurality of branch target addresses of previously executed branch instructions in a branch target address cache (BTAC) (fig. 2, BTB 56; col. 7, lines 40-42).
- b) Shiell has not taught caching a length of each of the branch instructions in the BTAC.

However, Hsu has taught a BTB which caches the length of each branch instruction. See column 13, lines 18-30. This allows the system to accommodate branch instructions that cross multiple cache blocks, which is useful in variable length instruction systems, as both Hsu and Shiell are (see the title of Hsu and column 6, lines 32-35 of Shiell). Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to cache the length of a branch instruction so that the system may accommodate branches that span multiple cache lines.

- c) accessing said BTAC with a fetch address of an instruction cache after said caching (fig. 2 indicates that the fetch address FA is used to access the BTB 56; col. 8, lines 13-16 indicate that the accessing is done when branch history is stored in the BTB).

d) determining whether said fetch address hits in said BTAC in response to said accessing (col. 8, lines 13-16).

e) branching the microprocessor to one of said plurality of branch target addresses selected by said fetch address if said fetch address hits in said BTAC whether or not a branch instruction is cached in a line of said instruction cache indexed by said fetch address (col. 7, lines 40-45 indicate that the target address is used to generate the fetch address during the fetch stage before the decoding the instruction without knowledge of whether the instruction at that fetch address in the instruction cache is still a branch or not i.e. speculative execution [col. 8, lines 13-16] because the BTB and the instruction cache are accessed in parallel using the fetch address FA).

f) Shiell has not taught storing a discrete indication that said branching was performed if said branching is performed, and determining from said discrete indication, subsequent to said storing, that said branching was performed. However, Black has taught such a concept. See column 6, lines 34-37, and note that the HIT?MISS signal is store din a decode buffer for later use. This HIT/MISS signal, when set to HIT, indicates that a branch has been detected and branching has been performed. This signal is then used to detect if a phantom branch was encountered, and if one has been encountered, then corrective measures are taken. As a result, in order to detect branching for a non-branch instruction, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to include Black's concept of storing an indication.

44. Referring to claim 67, Shiell in view of Hsu has taught a method as described in claim 66. Furthermore, claim 67 is rejected for the same reasons set forth in the rejection of claim 35.

45. Referring to claim 68, Shiell in view of Hsu has taught a method as described in claim 67. Furthermore, claim 68 is rejected for the same reasons set forth in the rejection of claim 36.

46. Referring to claim 70, Shiell in view of Hsu has taught a method as described in claim 69. Furthermore, claim 70 is rejected for the same reasons set forth in the rejection of claim 38.

47. Claims 43-44, and 46-49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Black, as applied above.

48. Referring to claim 43, Black has taught a microprocessor as described in claim 32.

a) Black has not explicitly taught instruction decode logic, coupled to said instruction buffer, for decoding said instructions to indicate which byte of each of said instructions is an opcode byte. However, Official Notice is taken that this is a common scheme that is well known and expected in the art. That is, in variable byte-length instruction systems, the system must determine which bytes are the opcodes so that it knows what instructions to perform. And, as mentioned, this allows a system to execute variable length instructions. Consequently, in order to make Black's system capable of executing variable length instructions, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Black to include the aforementioned decode logic.

b) Black has also not taught prediction check logic, coupled to receive said plurality of indicators from said instruction buffer, wherein if one of said plurality of indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said instructions, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if a same

byte of said one of said instructions is not indicated by said instruction decode logic and by said plurality of indicators stored in said instruction buffer to be said opcode byte. However, Official Notice is taken that it is common for caches (BTAC/BTB included) to be constructed such that multiple fetch addresses map to the same cache entry. If in the situation where a non-opcode byte is encountered and it maps to a branch entry that is allocated to a branch opcode, then a prediction should not be made because it is unknown whether this non-opcode byte is actually associated with a branch instruction. But, since it maps to the same location as a branch, the system will assume a branch has been encountered and an erroneous prediction would have been made. Such a cache construction is desired because it is not often that instructions map to the same location and it also keeps the cache smaller. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to reduce the size of the BTAC and determine erroneous predictions if a non-opcode byte results in a prediction being made.

49. Referring to claim 44, Black has taught a microprocessor as described in claim 31.

- a) Black has further taught instruction decode logic, coupled to said instruction buffer, for decoding said instructions to indicate whether each of said instructions is a non-branch instruction. See Fig.3, component 56, and note that clearly the system will differentiate branch and non-branch instructions as different operations must be performed based on the instruction type.
- b) Black has not taught prediction check logic, coupled to receive said plurality of indicators from said instruction buffer, wherein if one of said plurality of indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said instructions, said prediction check logic indicates that

the microprocessor erroneously branched to said one of said branch target addresses if said instruction decode logic indicates said one of said instructions is a non-branch instruction. However, Official Notice is taken that it is common for caches (BTAC/BTB included) to be constructed such that multiple fetch addresses map to the same cache entry. If in the situation where a non-branch instruction is encountered and it maps to a branch entry that is allocated to a branch instruction, then a prediction should not be made because non-branch instructions do not require predictions. But, since it maps to the same location as a branch, the system will assume a branch has been encountered and an erroneous prediction would have been made. Such a cache construction is desired because it is not often that instructions map to the same location and it also keeps the cache smaller. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to reduce the size of the BTAC and determine erroneous predictions if a non-branch instruction results in a prediction being made.

50. Referring to claim 46, Black has taught a microprocessor as described in claim 31. Furthermore, Black has taught prediction check logic, coupled to receive said plurality of indicators from said instruction buffer, wherein said one of said instructions is a branch instruction, wherein if one of said plurality of indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if a resolved direction of said branch instruction does not match a direction of said branch instruction predicted by said branch target address cache. This is deemed to be inherent because if the resolved direction of a branch

does not match the predicted direction, then a misprediction has occurred (erroneously branched).

51. Referring to claim 47, Black has taught a microprocessor as described in claim 31. Black has further taught prediction check logic, coupled to receive said plurality of indicators from said instruction buffer, wherein said one of said instructions is a branch instruction, wherein if one of said plurality of indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if a resolved target address of said branch instruction does not match said one of said branch target addresses to which the microprocessor speculatively branched. This is deemed to be inherent because if the resolved target address of a branch does not match the speculative target address, then a misprediction has occurred (erroneously branched).

52. Referring to claim 48, Black has taught a microprocessor as described in claim 31. Black has further taught a non-speculative branch predictor, coupled to instruction buffer, for generating a non-speculative predicted target address of a branch instruction for which said branch target address cache provided said one of said branch target addresses to which the microprocessor speculatively branched and branch control logic, coupled to receive said plurality of indicators from said instruction buffer, wherein if one of said plurality of indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said branch control logic causes the microprocessor to branch to said non-speculative predicted target address if said non-speculative predicted target address generated by said non-speculative branch predictor does not match said

one of said branch target addresses of said branch instruction provided by said branch target address cache. This is deemed to be inherent because if the resolved target address (non-speculative prediction) of a branch does not match the actual prediction from the BTAC, then a misprediction has occurred (erroneously branched), and the system will be corrected by branching to the non-speculative address.

53. Referring to claim 49, Black has taught a microprocessor as described in claim 31. Black has further taught a non-speculative branch predictor, coupled to instruction buffer, for generating a non-speculative predicted direction of a branch instruction for which said branch target address cache provided said one of said branch target addresses to which the microprocessor speculatively branched, and branch control logic, coupled to receive said plurality of indicators from said instruction buffer, wherein if one of said plurality of indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said branch control logic causes the microprocessor to branch to a next instruction sequential to said branch instruction if said non-speculative predicted direction generated by said non-speculative branch predictor is a not taken prediction. This is deemed to be inherent because if the resolved target address (non-speculative prediction or actual target address) of a branch does not match the actual prediction from the BTAC, then a misprediction has occurred (erroneously branched), and the system will be corrected by branching to the non-speculative address. When the actual target address is the next sequential instruction, then the microprocessor will begin fetching from the next sequential address.

54. Claim 45 is rejected under 35 U.S.C. 103(a) as being unpatentable over Black, as applied above, in view of Stiles, U.S. Patent No. 5,513,330, and further in view of Sinharoy, U.S. Patent No. 6,457,120.

55. Referring to claim 45, Black has taught a microprocessor as described in claim 31.

a) Black has further taught instruction decode logic, coupled to said instruction buffer. See Fig.3.

b) Black has not taught that the decode logic determines a length of each of said instructions, and prediction check logic, coupled to receive said plurality of indicators from said instruction buffer, wherein if one of said plurality of indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said instructions, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if said length received from said instruction decode logic does not match a speculative length of said one of said instructions provided by said branch target address cache. However, Stiles has taught a branch prediction cache which supplies target instruction lengths so that alignment and fetching may be done efficiently. See the first paragraph in the “Summary of Invention” section. Stiles has also taught calculating the lengths of target instructions which are not yet registered in the branch cache. See Fig.2 and Fig.4. Although Stiles has not taught comparing the predicted and calculated addresses, a person of ordinary skill in the art would have recognized that the comparison would be useful in systems which experience aliasing (where multiple branches map to the same entry in cache). Sinharoy has taught such a system in column 4, lines 45-57. Basically, aliasing is sometimes allowed so that the cache may be kept smaller (there will not be

an individual entry for each instruction). The hope is that it isn't often that instructions map to the same location, and so the reduction in cache size is worth the occasional penalty of an instruction mapping to a shared location. As a result it would have been obvious to one of ordinary skill in the art to implement aliasing in Black. However, with aliasing implemented, a branch might map to an address which belongs to another branch, and consequently, the target instructions and their lengths may be incorrect. Therefore, a misprediction can be determined by comparing the predicted addresses and the calculated addresses for the actual instructions. Therefore, in order to detect mispredictions in a system with aliasing, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Black in view of Sinharoy to include a comparison of predicted and actual instruction lengths.

56. Claims 56-58 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell in view of Hsu, as applied above, and further in view of Bae et al., U.S. Patent No. 6,044,459 (as applied in the previous Office Action and herein referred to as Bae).

57. Referring to claim 56, Shiell in view of Hsu has taught a BTAC as described in claim 51. Furthermore, although Shiell verifies whether the target address was correct by comparing it with the actual next instruction address determined by the execution unit (col. 2, lines 48-51), Shiell has not taught that said portion of said speculative branch information comprises an indication of whether said one of said target addresses is a valid target address. However, Bae has taught a BTB entry format with a valid bit for indicating whether the target address is a valid target address (col. 5, lines 27-34). The valid bit is checked before providing the target address as the fetch address so that an invalid target is not incorrectly provided. It would have been obvious to

one of ordinary skill in the art at the time of the invention to recognize to add a valid bit in the BTB of Shiell indicating the validity of the target address so that an invalid target is not provided. One would have been motivated to do so because by including the valid bit, incorrect targets are not fetched and subsequent flushing of the pipeline is not required, thereby leading to improved performance.

58. Referring to claim 57, Shiell in view of Hsu has taught a BTAC as described in claim 56. Shiell in view of Hsu and further in view of Bae has further taught that said indication is populated to indicate said one of said target addresses is a valid target address in response to execution of the presumed branch instruction, wherein said one of said target addresses is resolved. This limitation is deemed inherent because the valid bit can be set to indicate the validity of the predicted target address only after the branch instruction is executed and the target is calculated.

59. Referring to claim 58, Shiell in view of Hsu has taught a BTAC as described in claim 56. Shiell in view of Hsu and further in view of Bae has further taught that said indication is populated to indicate said one of said target addresses is not a valid target address in response to detecting said one of said target addresses is erroneous subsequent to said providing said one of said target addresses on said output (This limitation is deemed inherent because the valid bit can be set to indicate that the predicted target address is invalid only after comparing the predicted target address in the BTB with the target address calculated on execution i.e. detecting that the target address is erroneous).

Response to Arguments

60. Applicant's arguments filed on November 29, 2004, have been fully considered but they are not persuasive.

61. Applicant argues the novelty/rejection of claim 50 on page 15 of the remarks, in substance that:

"Applicant respectfully asserts that Hsu does not teach a BTAC that stores a direct indication of whether a branch instruction presumed present in the cache line spans more than one line in the instruction cache, as recited in amended claim 50."

62. These arguments are not found persuasive for the following reasons:

a) As described in the new rejection of claim 50 above, Hsu has taught a BTB which caches the length of each branch instruction. The length is used to determine whether or not a branch spans a cache line by adding it to the branch instruction address. See column 13, lines 18-30. The examiner asserts that the length is a direct indication **with respect to the instruction address**. That is, given the instruction address, all that is needed is the length to make a determination as to whether an associated instructions spans cache lines. For instance, if a cache line holds four bytes, and a branch instruction begins in the first byte, a length of two directly indicates that the instruction does not span multiple cache lines. On the other hand, if the branch instruction begins in the last cache line byte, a length of two directly indicates that it does span multiple cache lines. With the instruction address handy, no additional information is needed other than the length. It should be realized that applicant has not defined in the claims what is meant by a direct indication.

63. Applicant argues the novelty/rejection of claims 43-44 on pages 19-20 of the remarks, in substance that:

Art Unit: 2183

"Applicant respectfully requests the Examiner to cite art in which a cache indicates a hit for multiple fetch addresses that map to the same entry. Otherwise, Applicant requests the Examiner withdraw the rejections."

64. These arguments are not found persuasive for the following reasons:

a) The examiner would like to bring Sinharoy, U.S. Patent No. 6,457,120, to applicant's attention. This reference is cited as extrinsic evidence for showing that multiple instructions may map to a single entry in a cache, and more specifically, multiple branch instructions may map to a single entry in cache. See column 4, lines 45-57 of Sinharoy, and note that he refers to a concept known as aliasing where multiple instructions may map to a given address. Basically, aliasing is sometimes allowed so that the cache may be kept smaller (there will not be an individual entry for each instruction). The hope is that it isn't often that instructions map to the same location, and so the reduction in cache size is worth the occasional penalty of an instruction mapping to a shared location.

65. Applicant argues the novelty/rejection of claims 43-44 on page 20 of the remarks, in substance that:

"Examiner has not taken Official Notice that the limitations themselves are common or well-known. Instead, the Examiner has asserted that the limitations added by claims 43 and 44 would have been obvious to one of ordinary skill in the art at the time of the invention, without citing a piece of art or taking Official Notice of the added limitations."

66. These arguments are not found persuasive for the following reasons:

a) The examiner pointed out that the decode logic of claims 43-44 was taught in Fig. 1 of Shiell, but each claim as a whole, Shiell did not teach. Consequently, Official Notice was taken. In addition, since applicant amended (and changed scope) of the claim on which these dependent claims depend, this argument is moot.

67. Applicant also argues the rejection of claim 45. However, since applicant amended (and changed scope) of the claim on which claim 45 depends, this argument is moot.

Conclusion

68. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

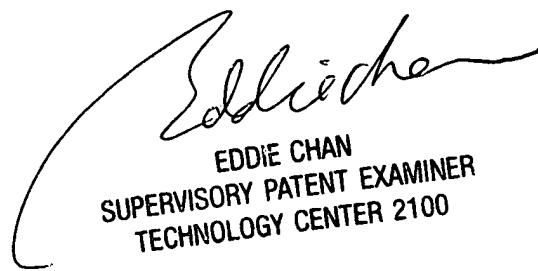
Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2183

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
January 14, 2005



EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100